

Figure 1: WebGuard Remote Administrative Architecture

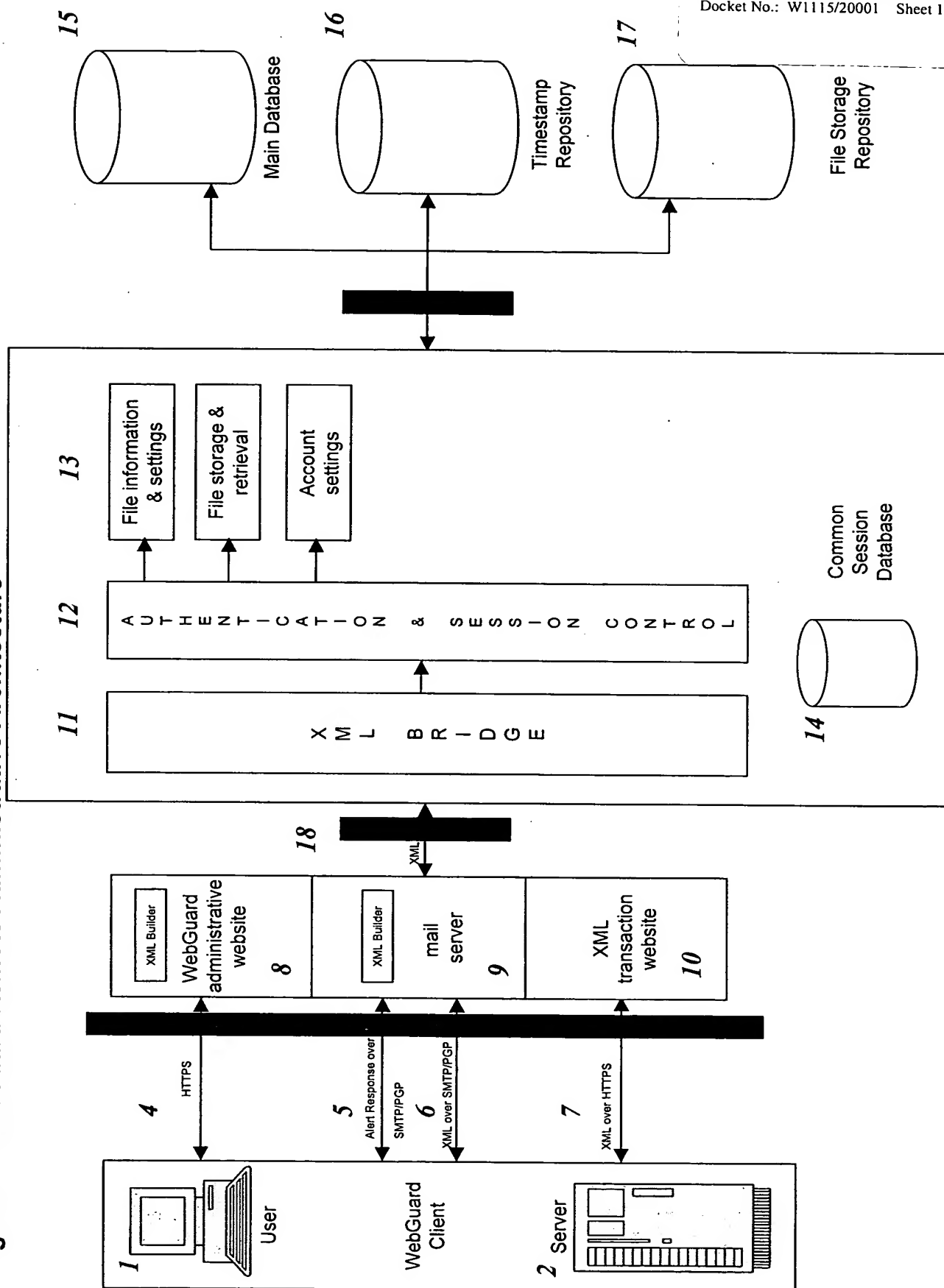


Figure 2: Simple Remote Process Overview

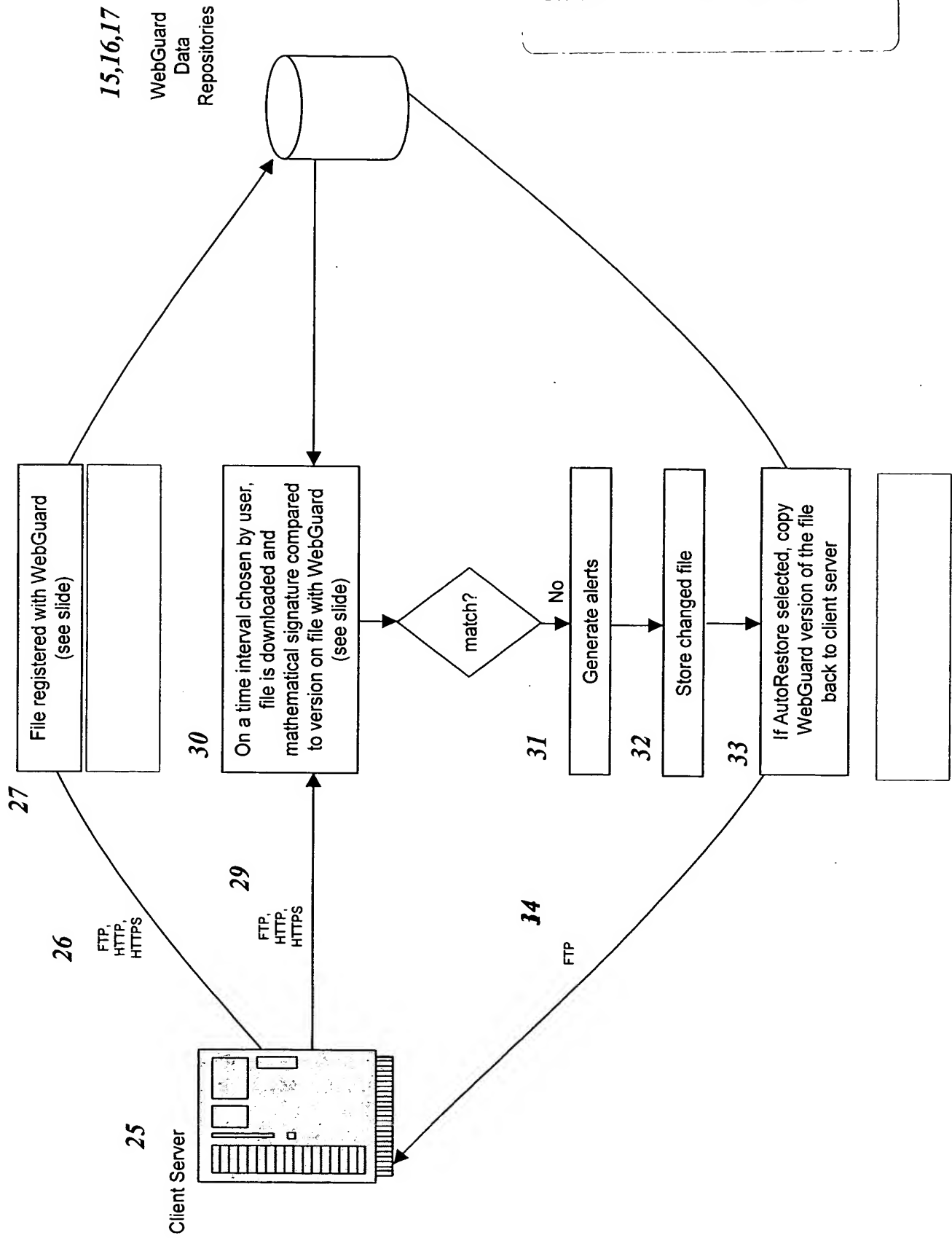


Figure 3: File Registration

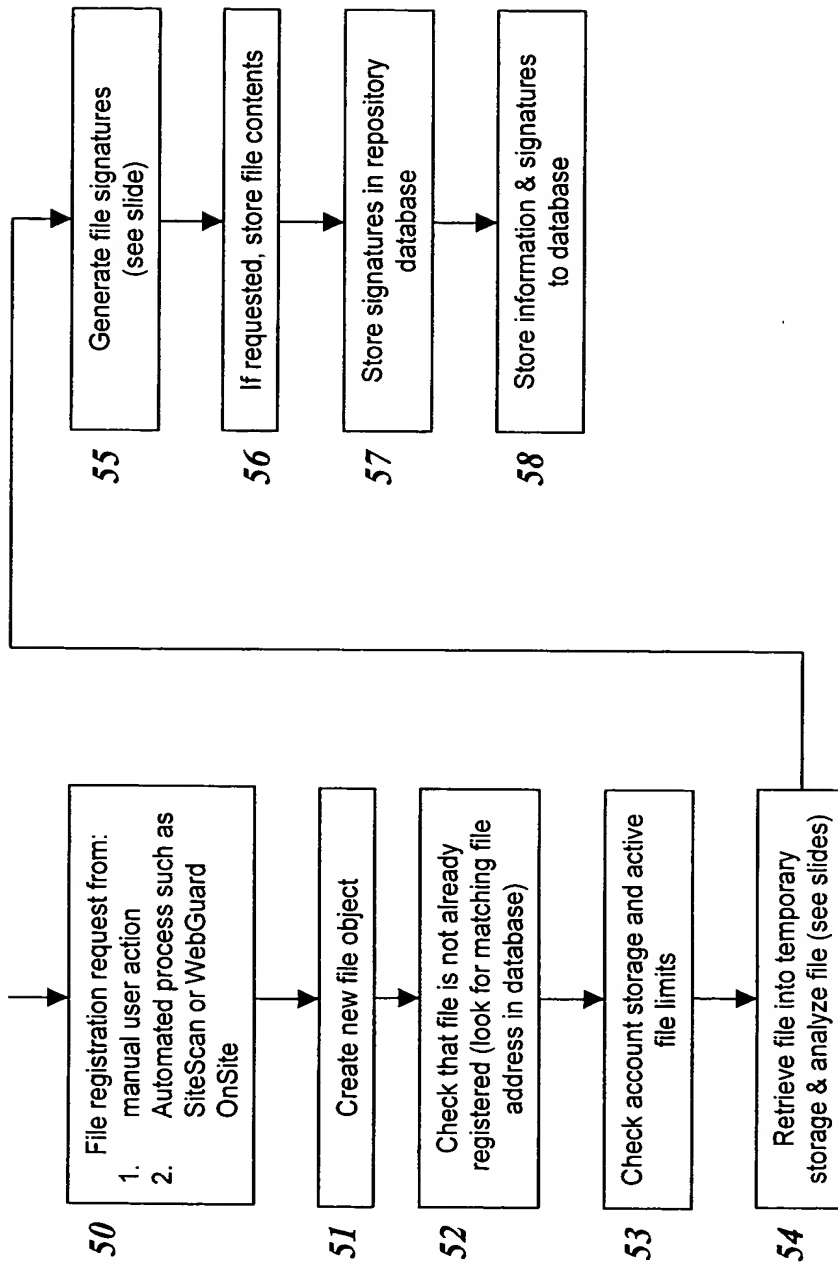


Figure 4: HTTP(S) File Retrieval

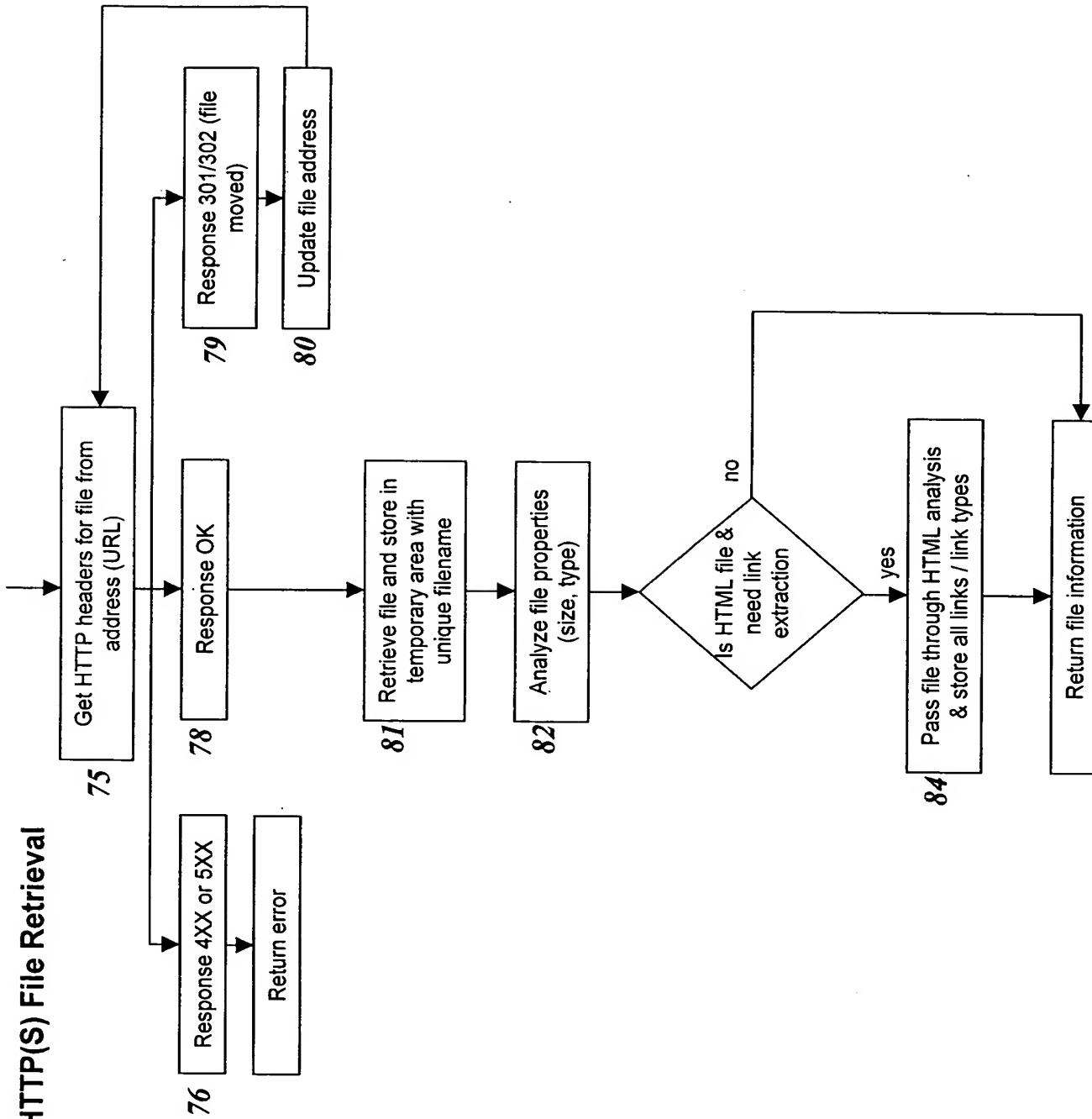
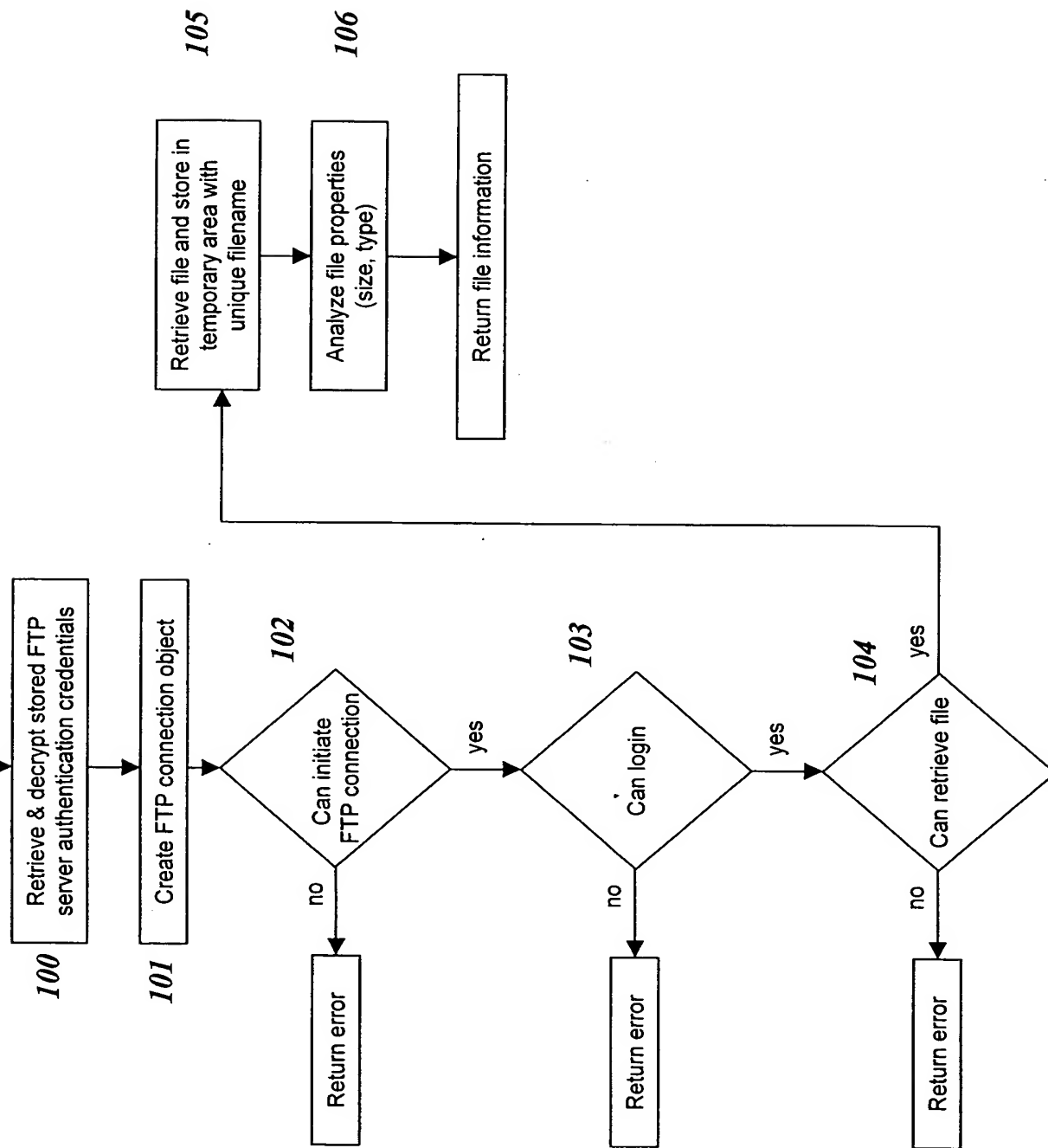
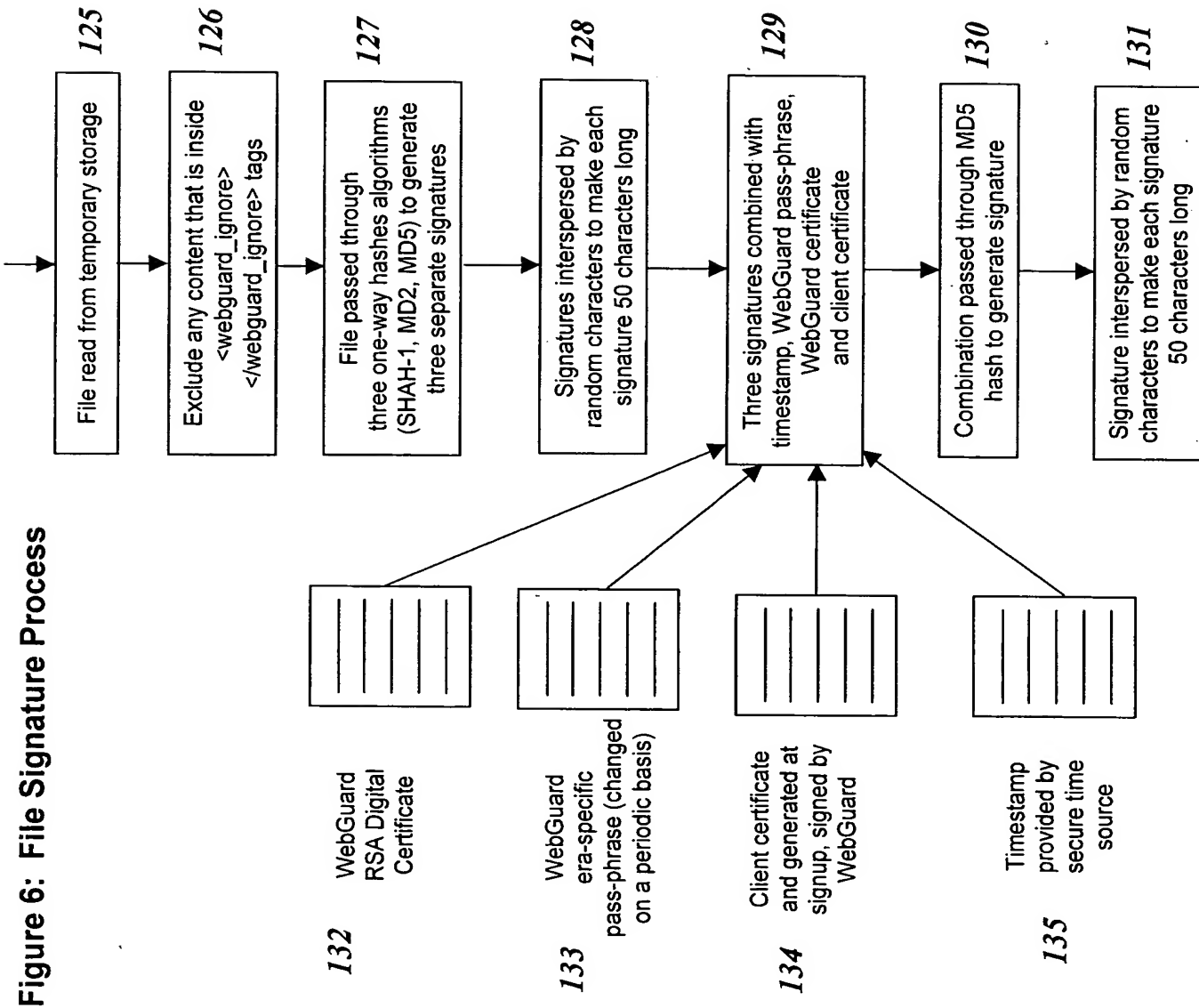


Figure 5: FTP File Retrieval





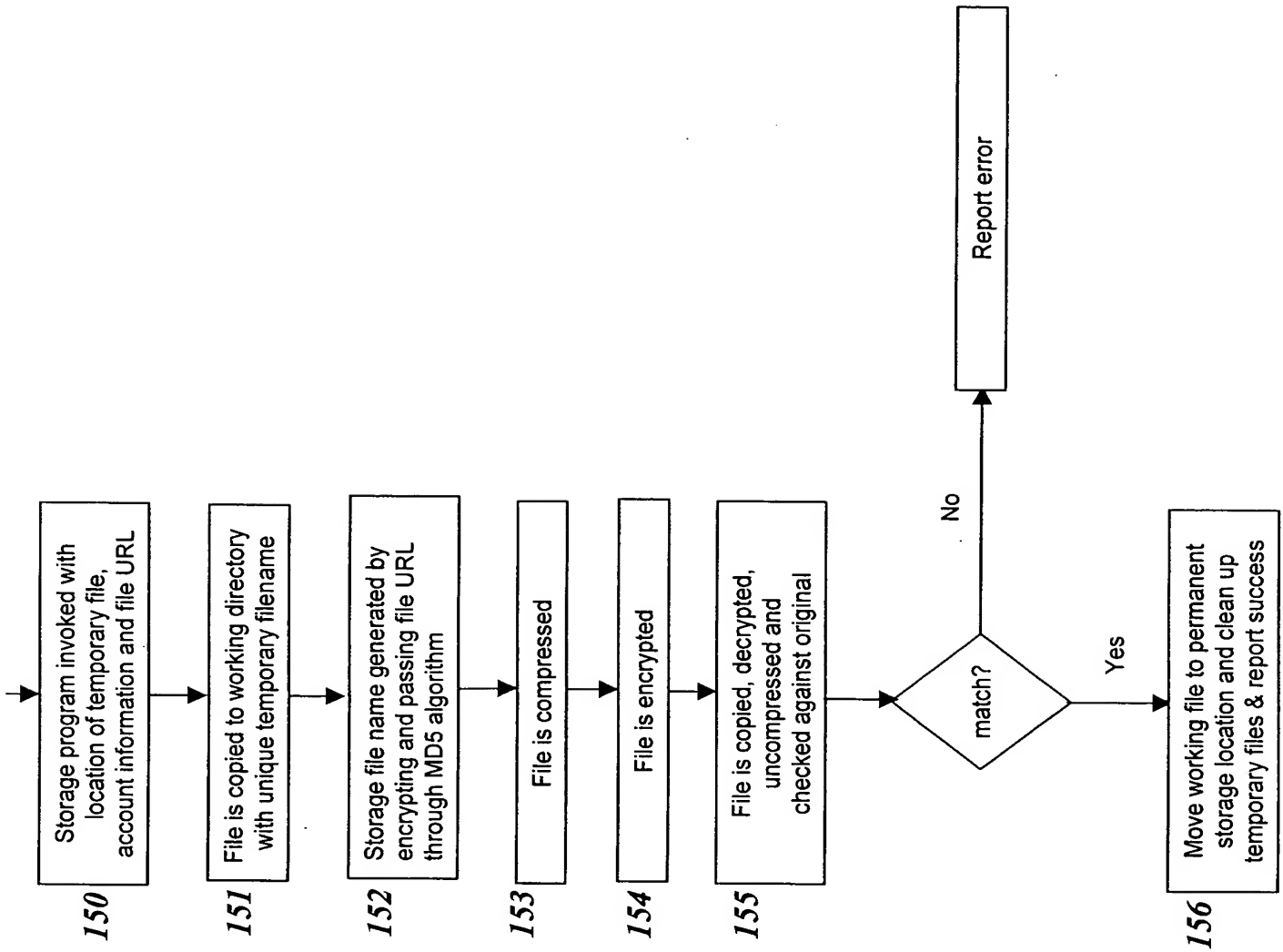


Figure 7: File Storage

Figure 8: HTTP/HTTPS File Scan Launcher

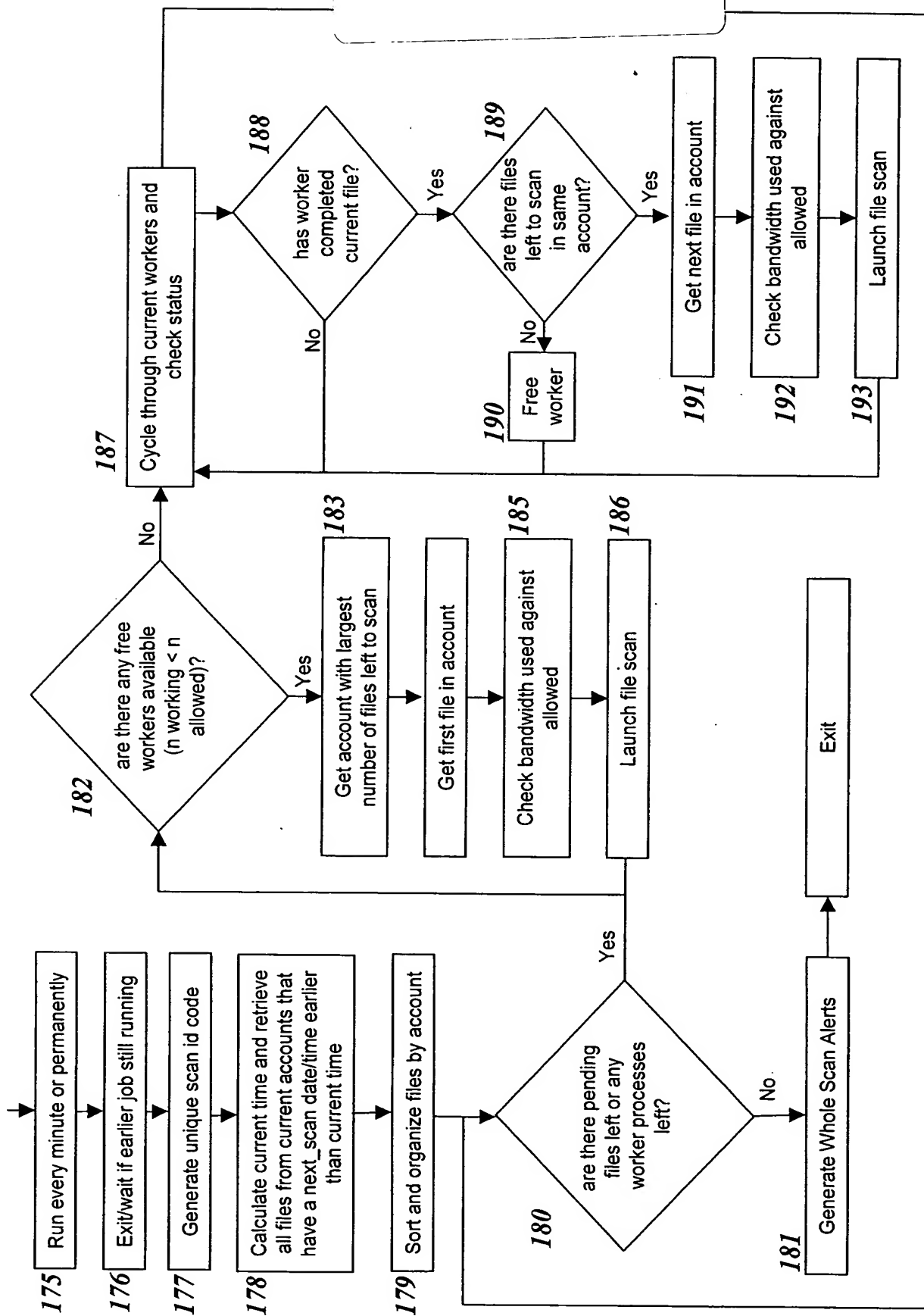


Figure 9: FTP File Scan Launcher

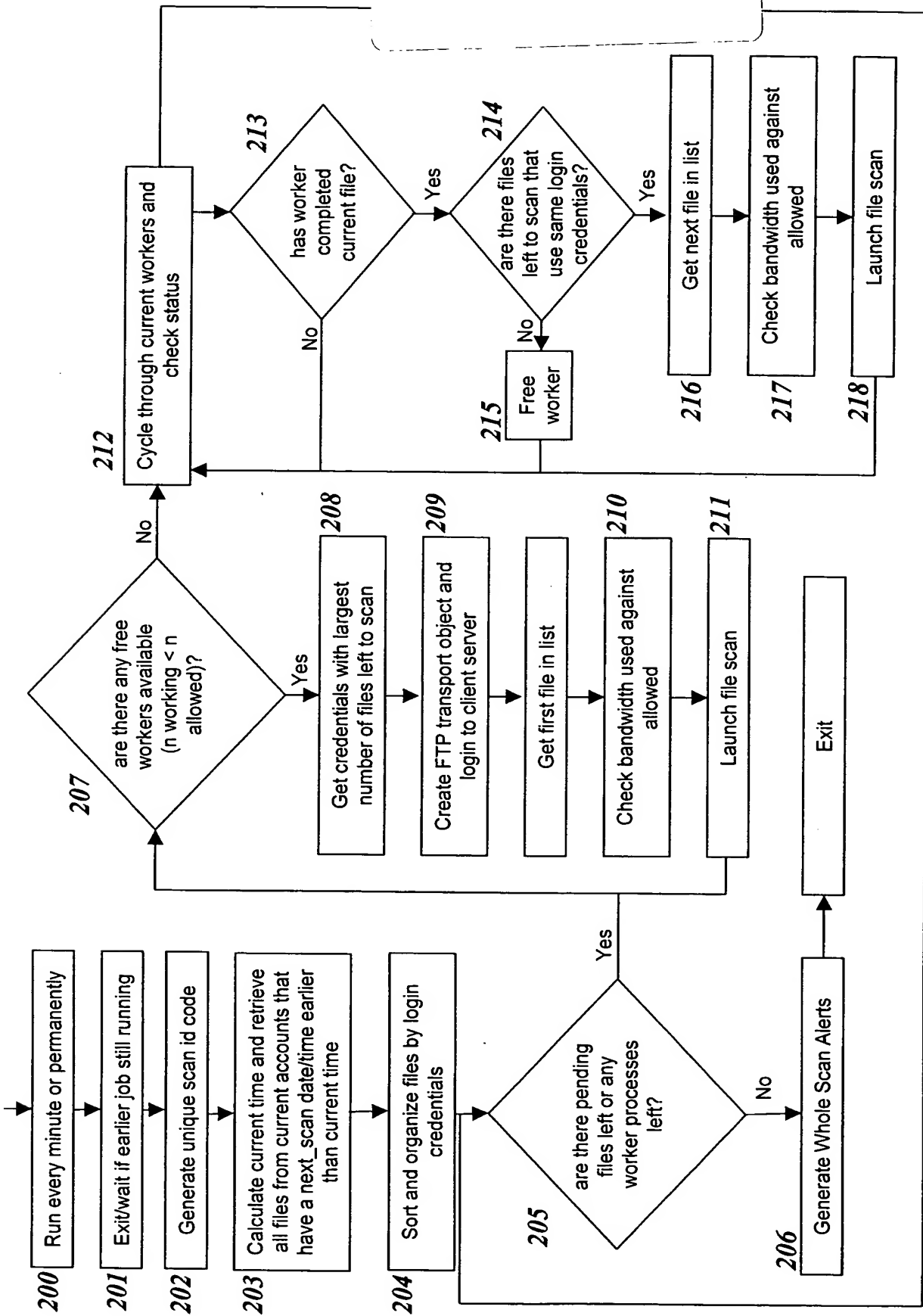


Figure 10: File Scan

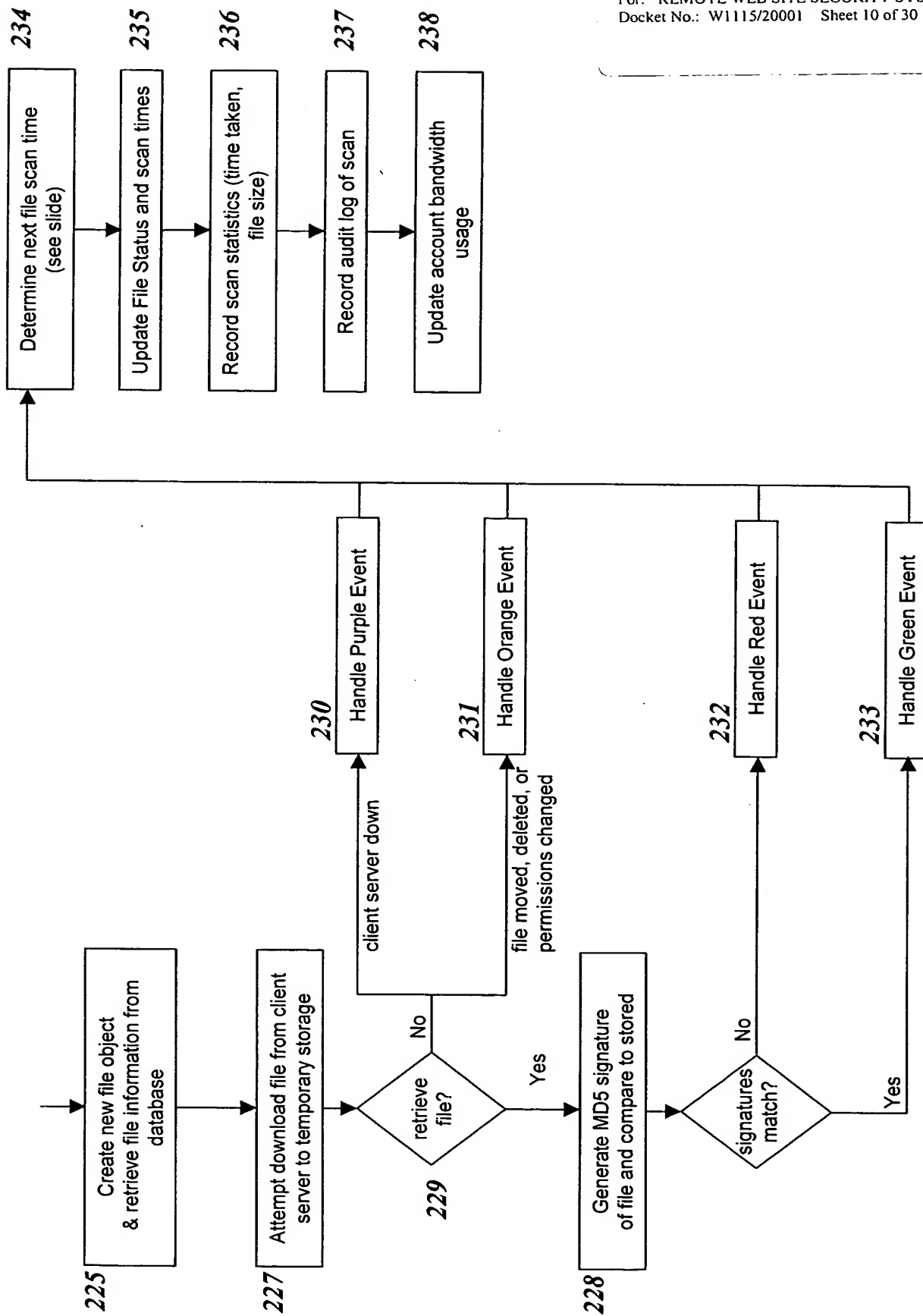


Figure 11: Determining Next File Scan Time

- WebGuard allows users to choose the scan interval (e.g. every 30 minutes, 1 day, 12 hours) for each file but does not allow users to choose the time of day that each file is scanned. WebGuard reserves the right to assign those times in order to ensure efficient bandwidth usage.
- Each file is assigned a 'base time', which is a number of minutes past midnight. The next scan time is then calculated from the base time and the scan interval. i.e. if the base time is 30 minutes and the scan interval is 60 minutes, scans will happen at 12:30, 1:30, 2:30, 3:30 etc. This allows WebGuard to spread bandwidth usage and prevents creep in scan times if the file takes longer than anticipated to scan. i.e. if we simply used the scan interval without the base time, we could be running scans at 12:30, 1:31, 2:32, 3:33 etc.
- Times are calculated in Unix time (seconds since January 1st 1970 00:00:00)

Step 1 – determine base_time_today for this file's base time

$\text{base_time_today} = \text{time_at_last_midnight} + \text{base_time_in_seconds}$

time_at_last_midnight is the time in seconds at midnight preceding the start of the scan
base_time_in_seconds is the base time for the file in seconds after midnight

Step 2 – determine the next scan time in Unix time

$\text{next_scan_time} = \text{base_time_today} + ((\text{integer}(\text{scan_start_time} - \text{base_time_today} / \text{scan_interval_in_seconds}) + 1) * \text{scan_interval_in_seconds})$

scan_start_time is the start of the scan in Unix time

scan_interval_in_seconds is the scan interval chosen by the user in seconds

Figure 12: Event Handler

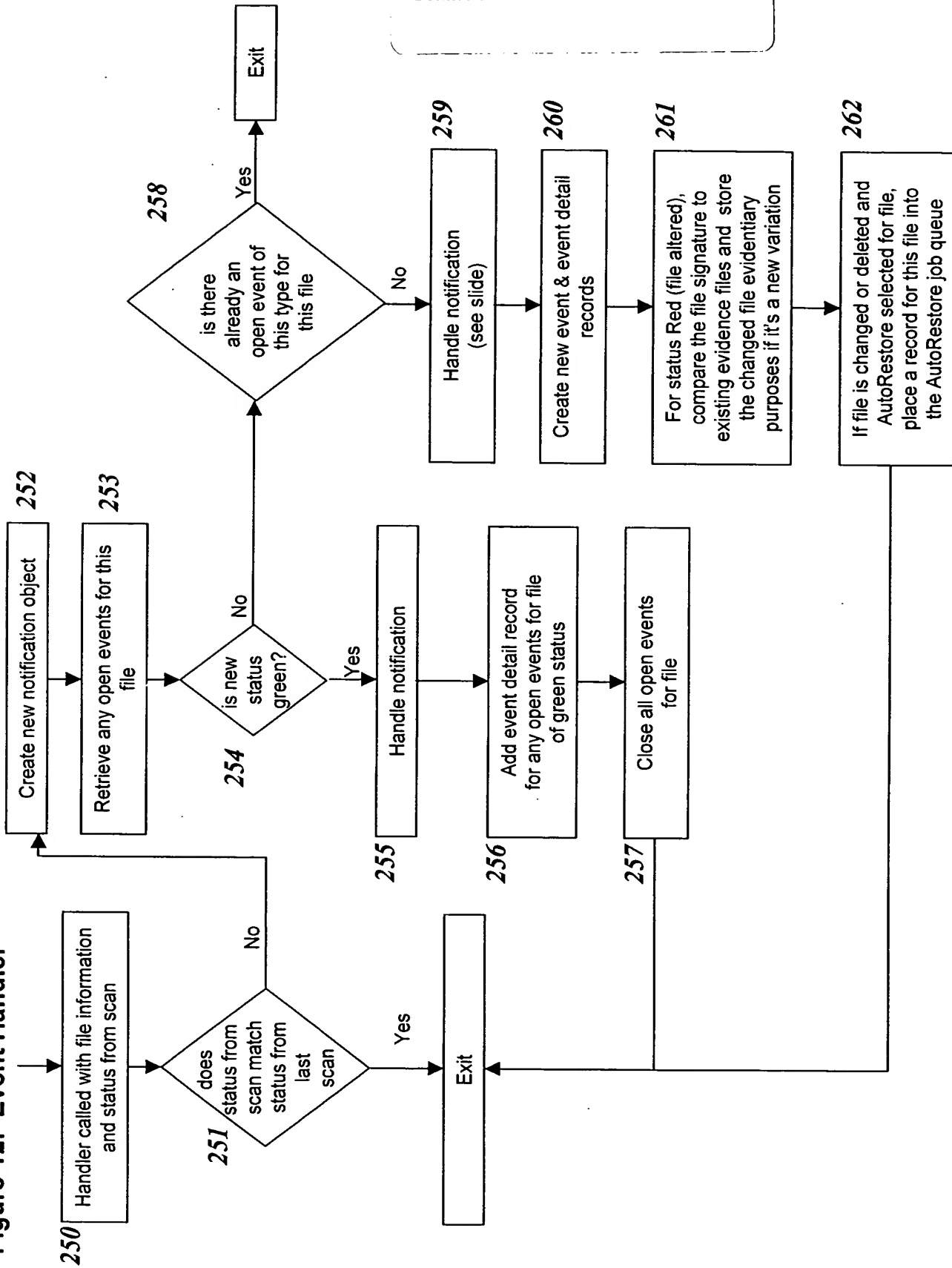


Figure 13: Notification Handler

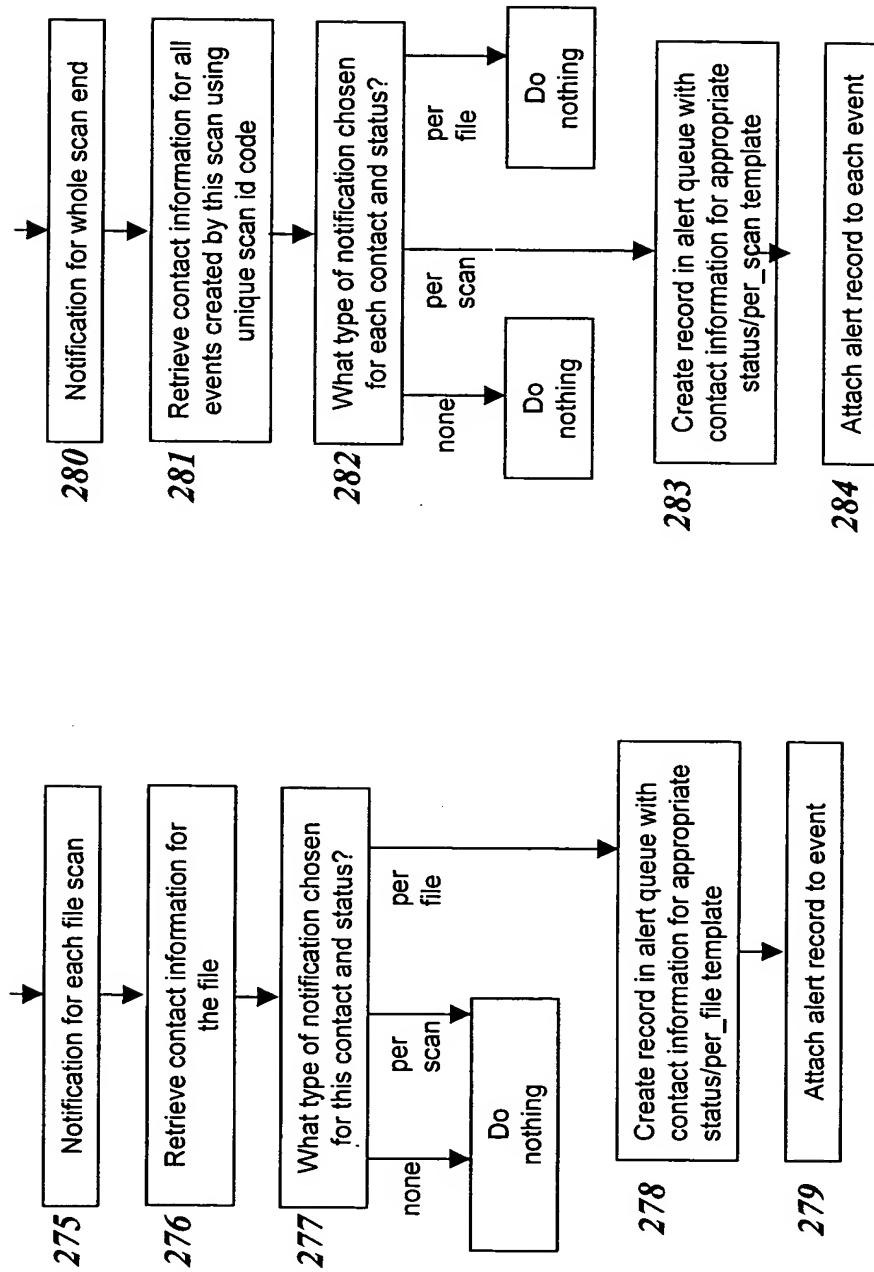


Figure 14: AutoRestore

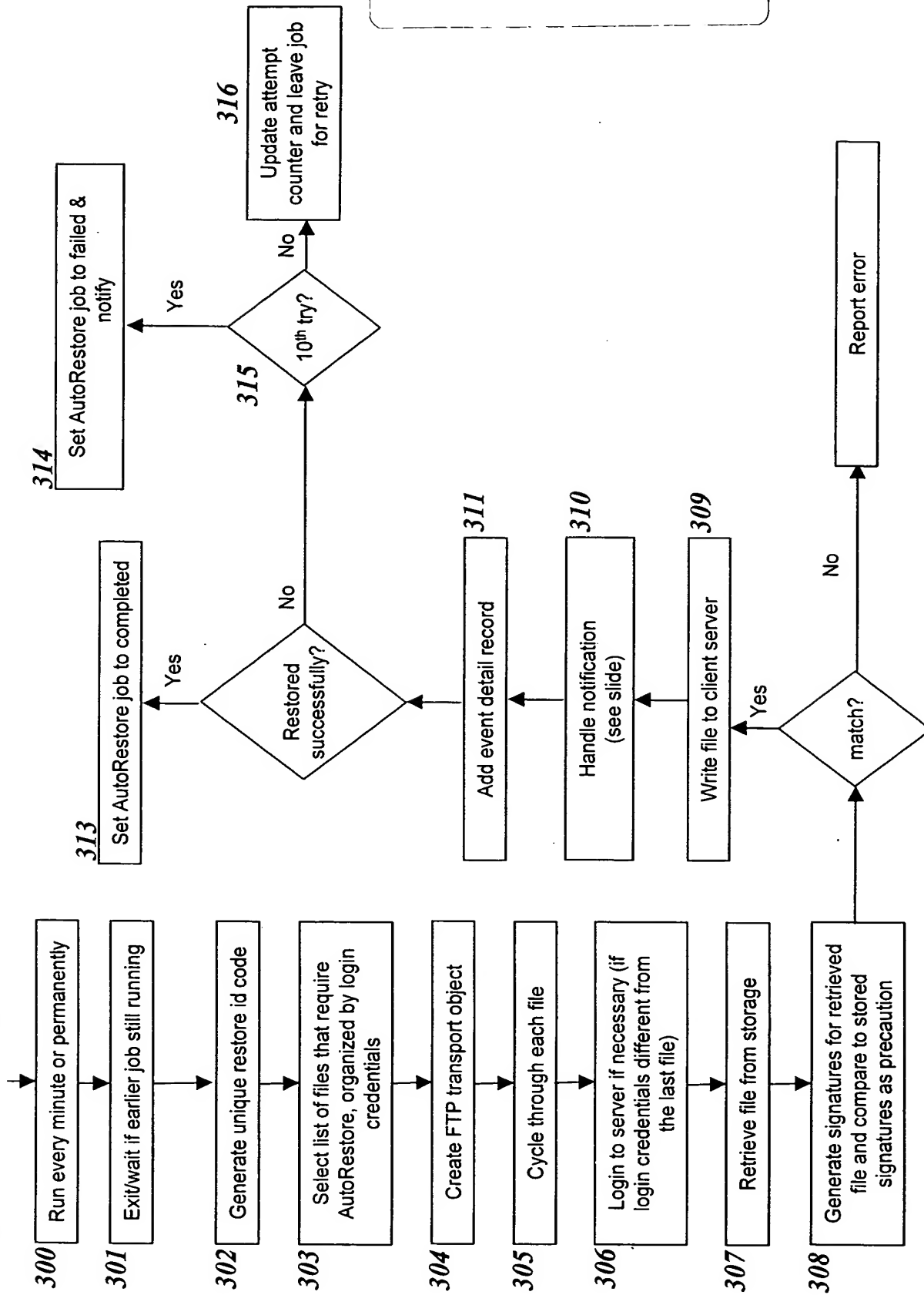


Figure 15: SiteScan Launcher

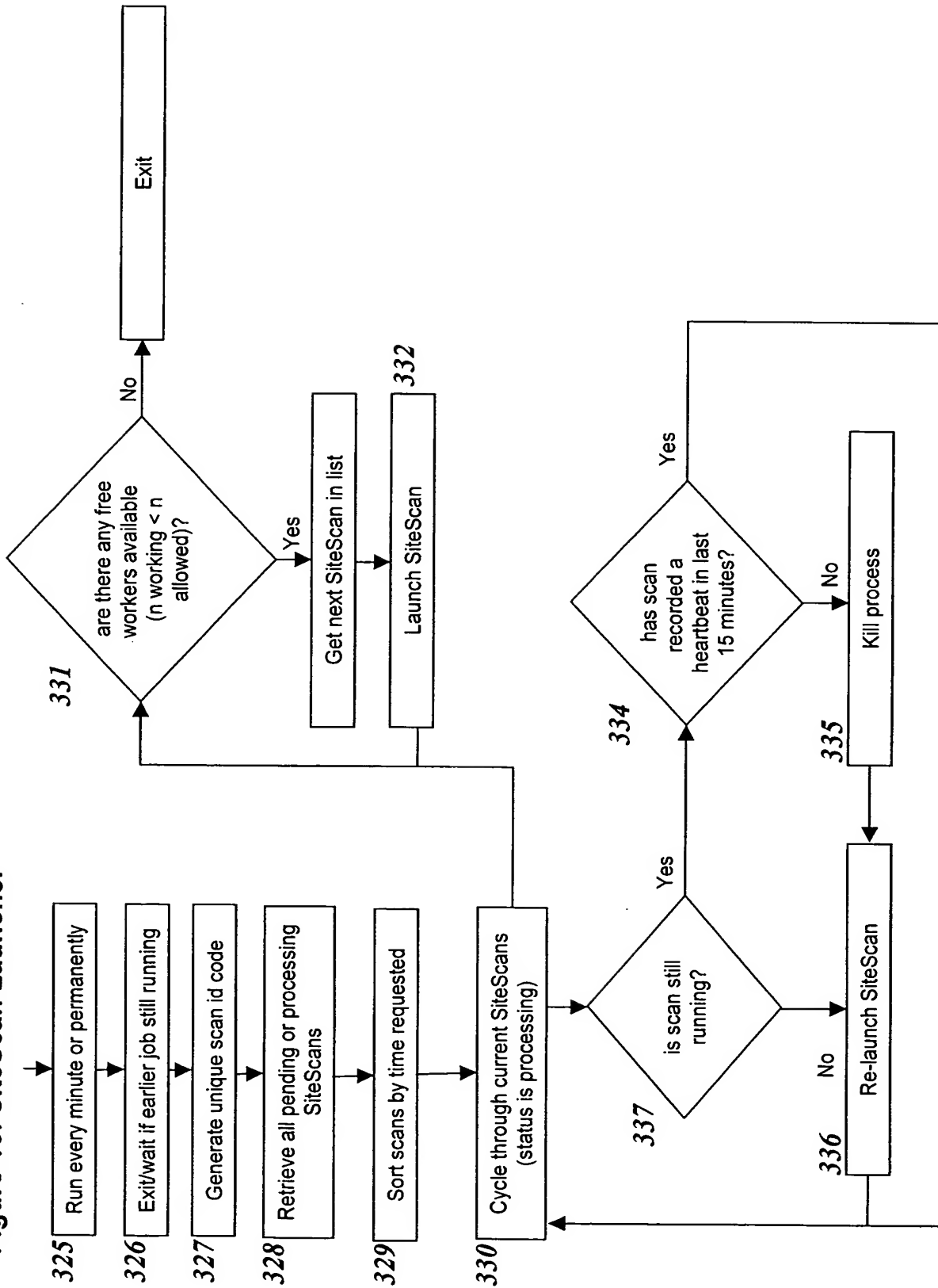


Figure 16: HTTP/HTTPS SiteScan

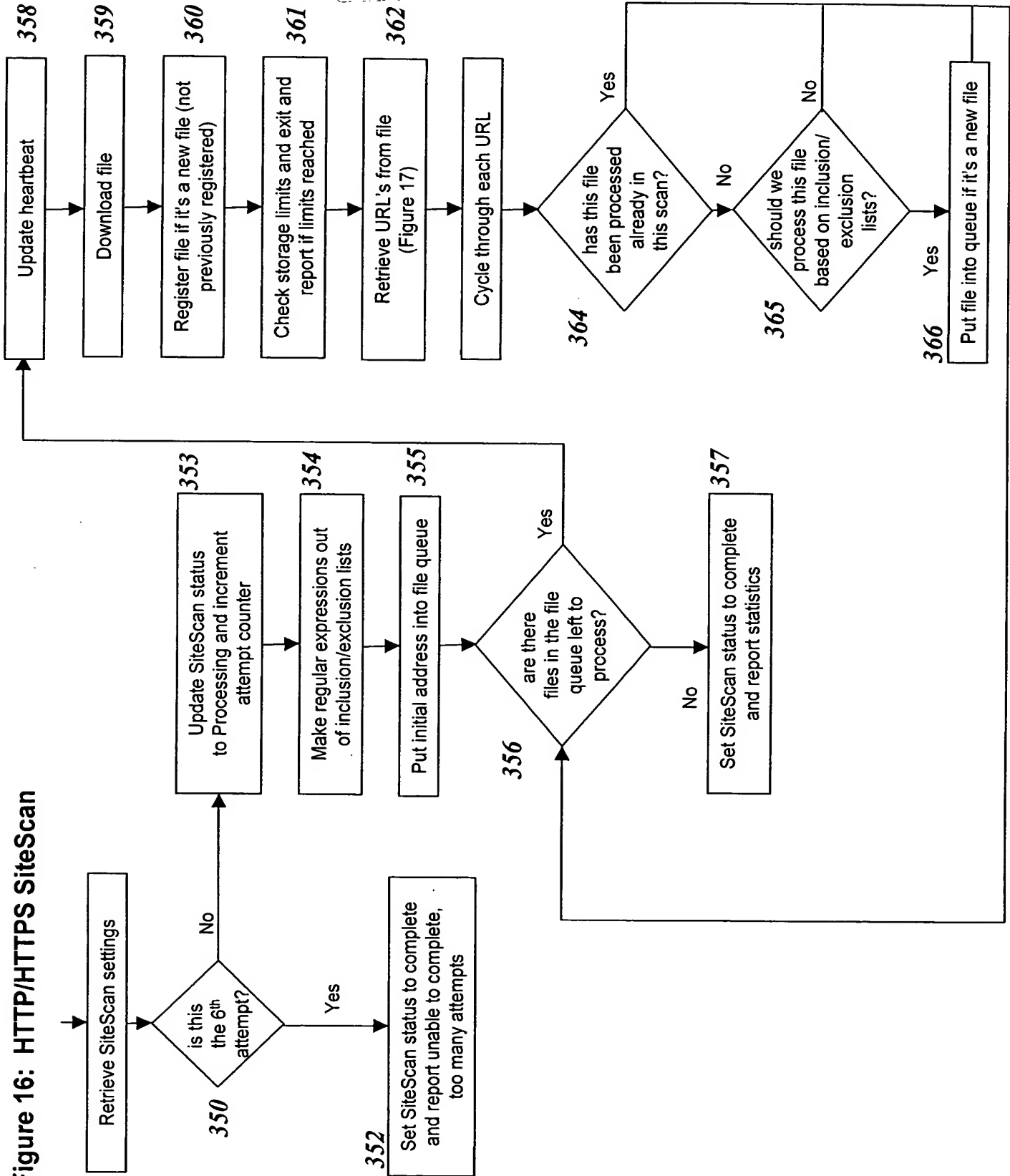


Figure 17: Process a File to Extract URL's

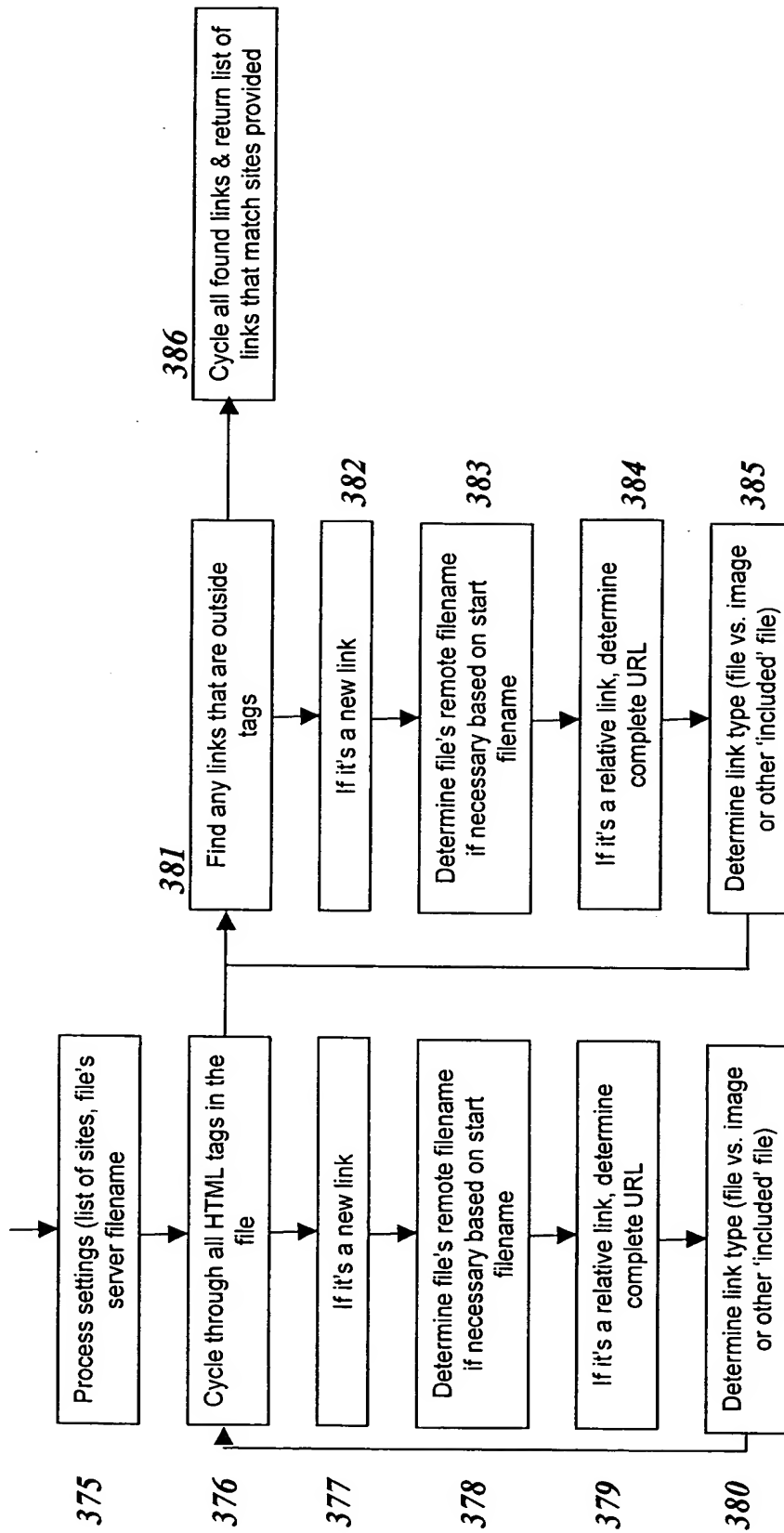


Figure 18: FTP SiteScan

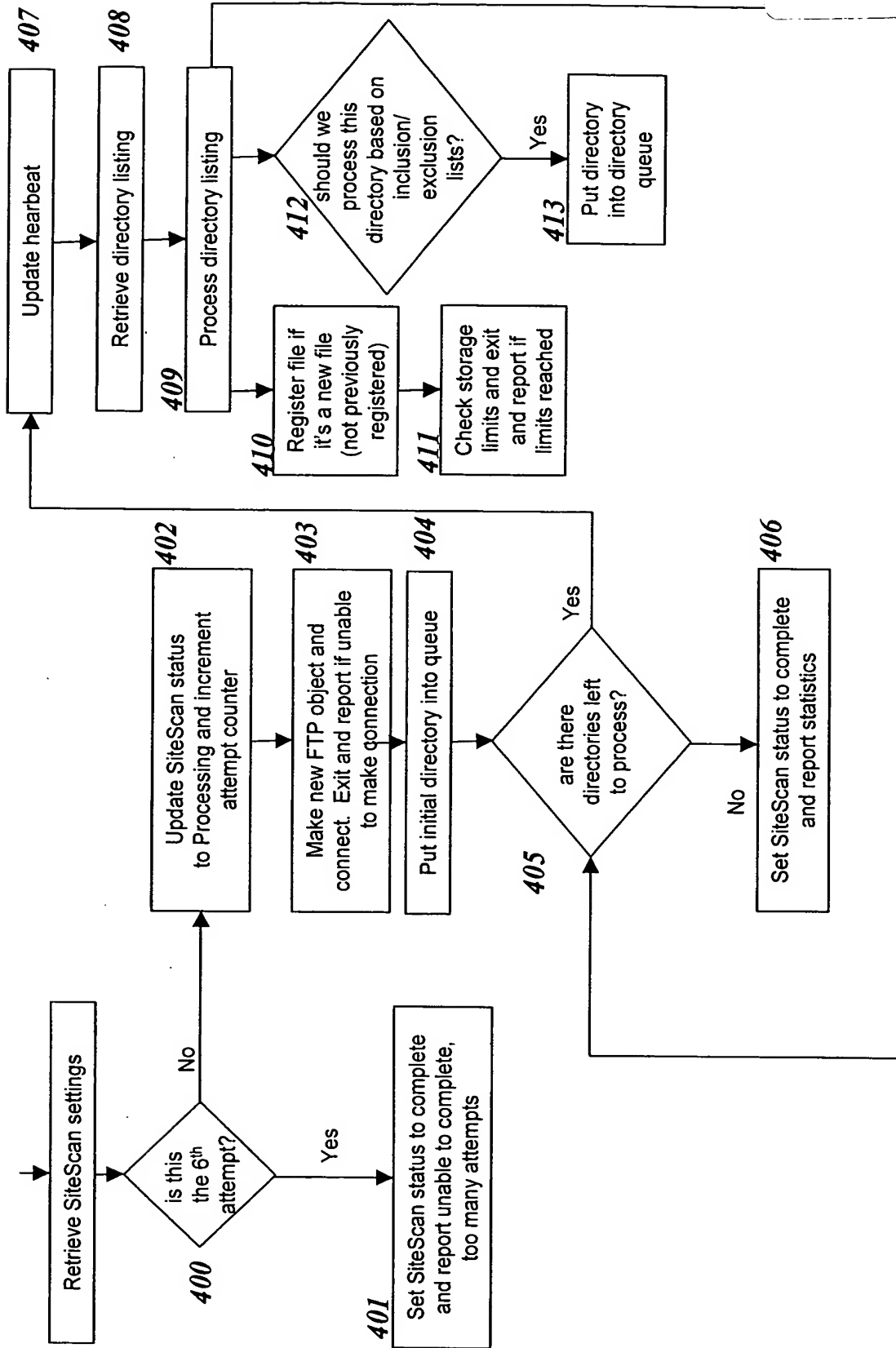


Figure 19: Scan Log Compression

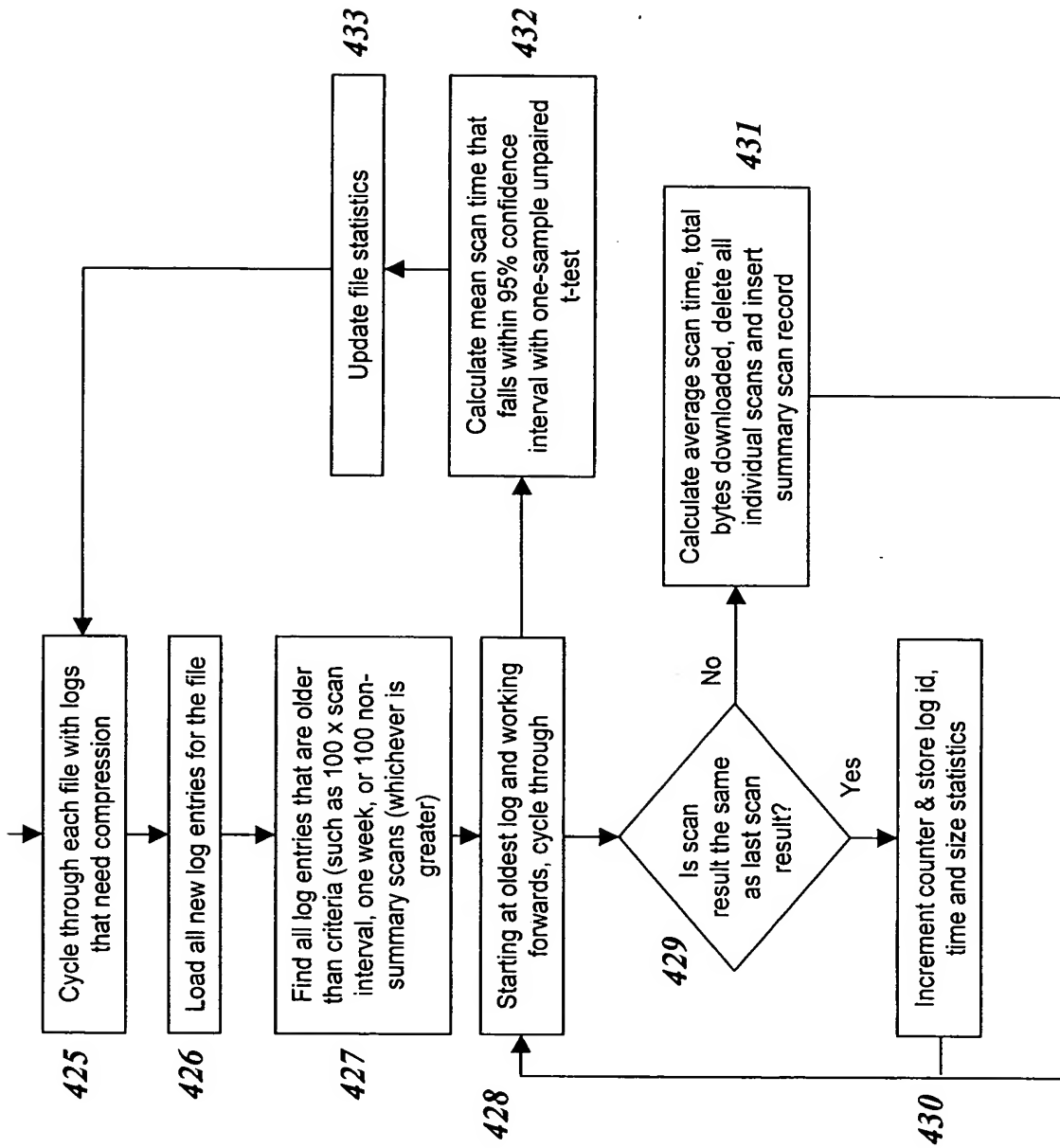


Figure 20: File Base-Time Assignment

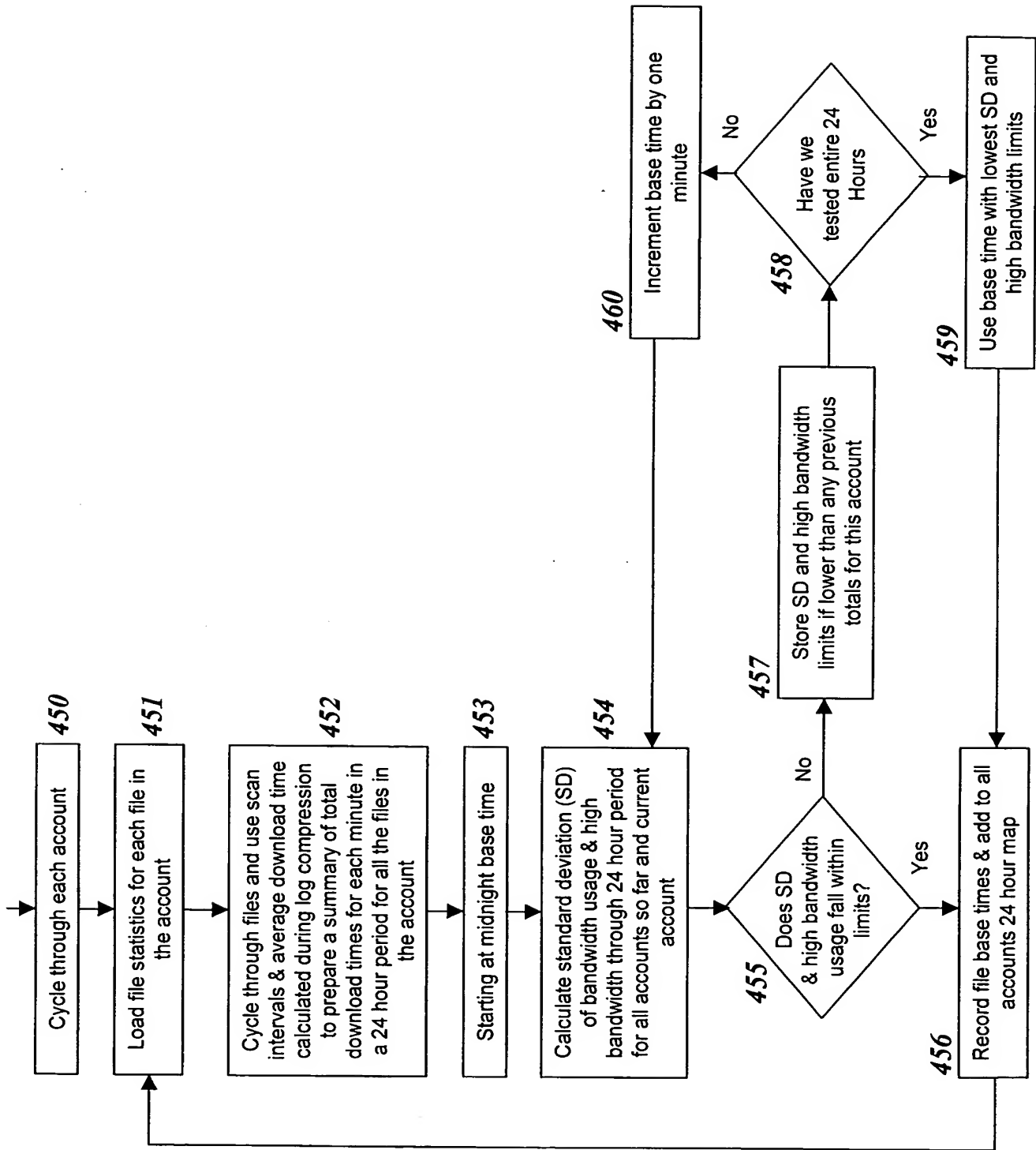


Figure 21: XML Bridge (using XML-RPC Spec)

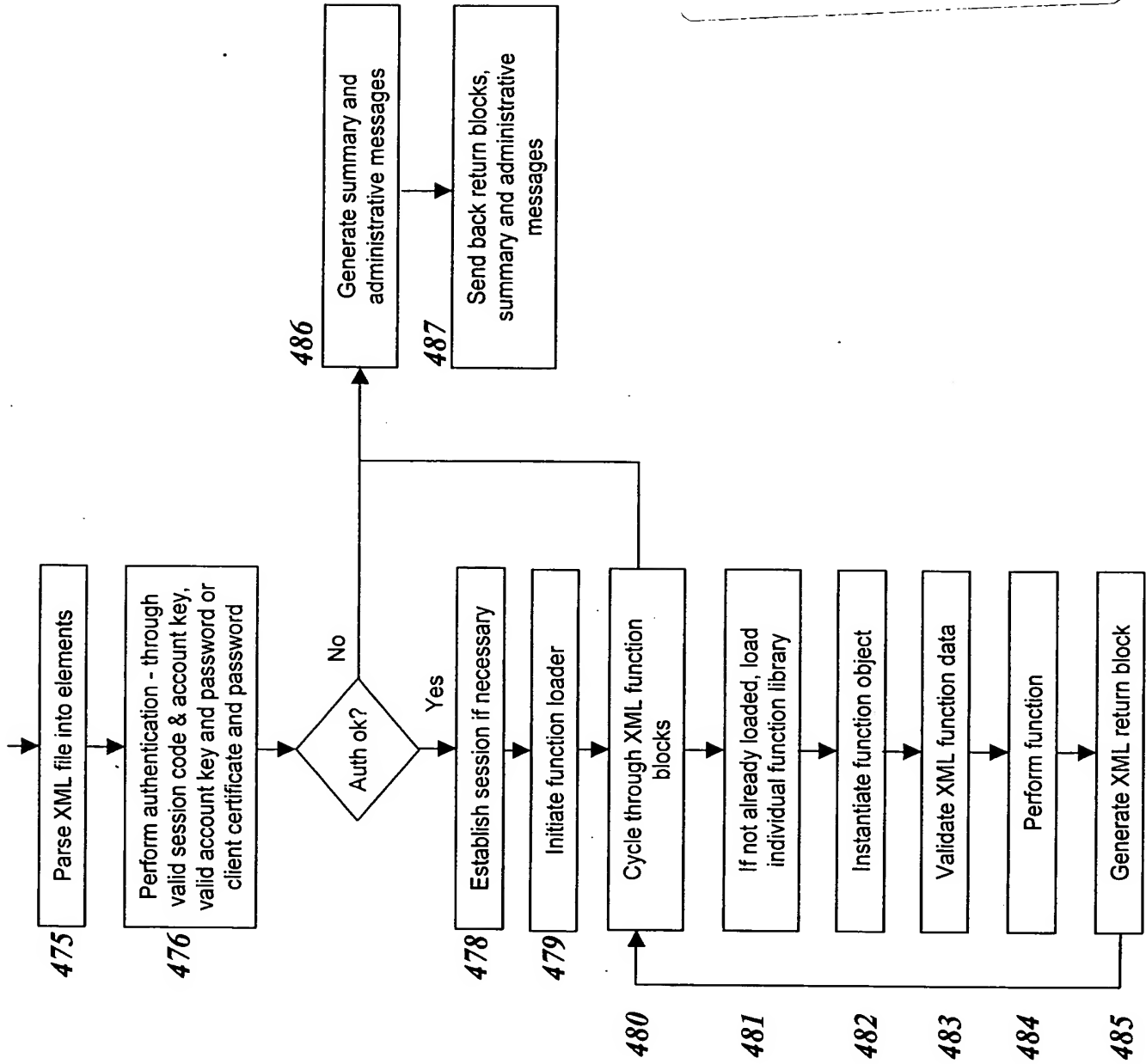


Figure 23: WebGuard OnSite Architecture

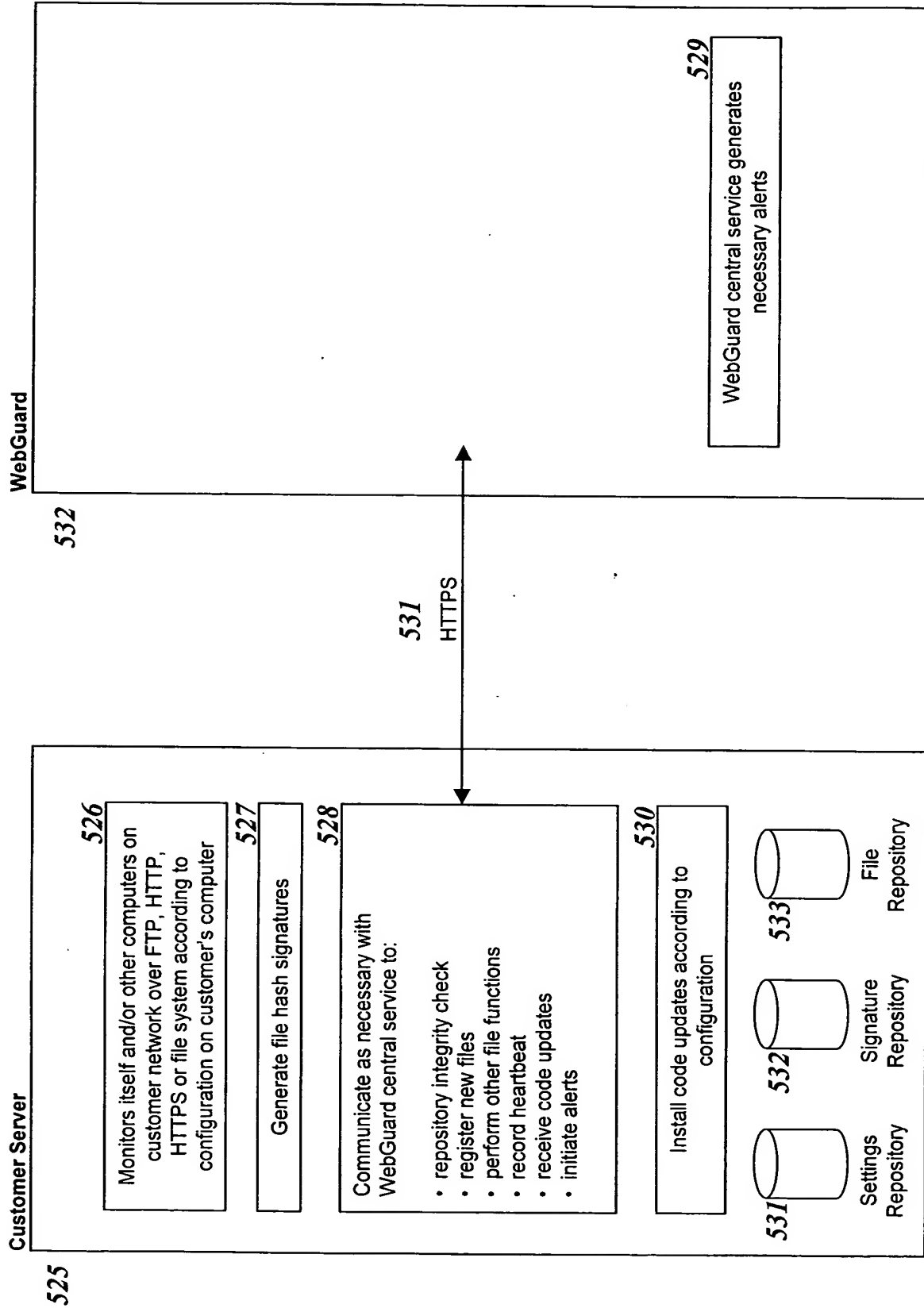


Figure 24: OnSite File Scheduling (on customer server)

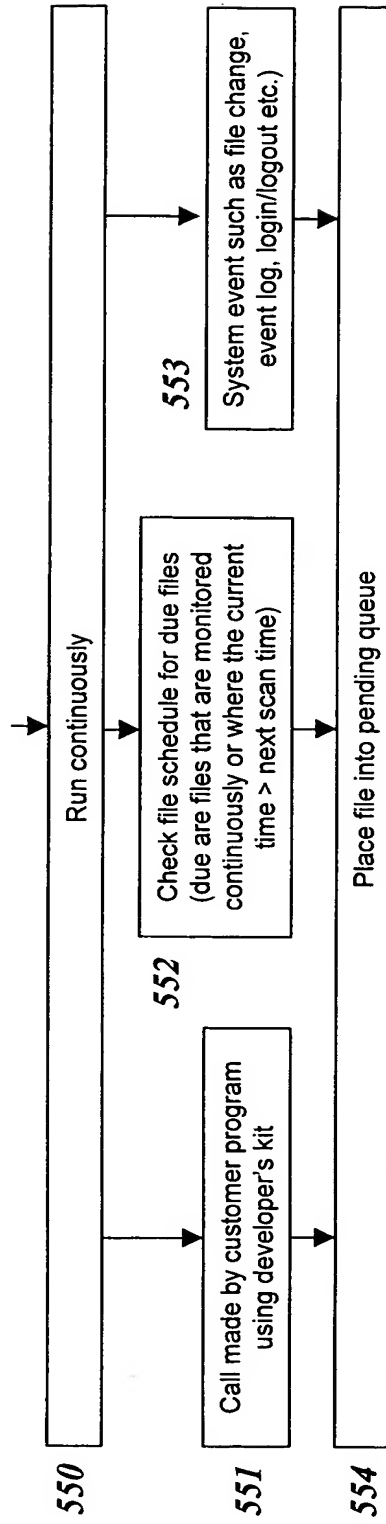


Figure 25: OnSite File Scanning (on customer server)

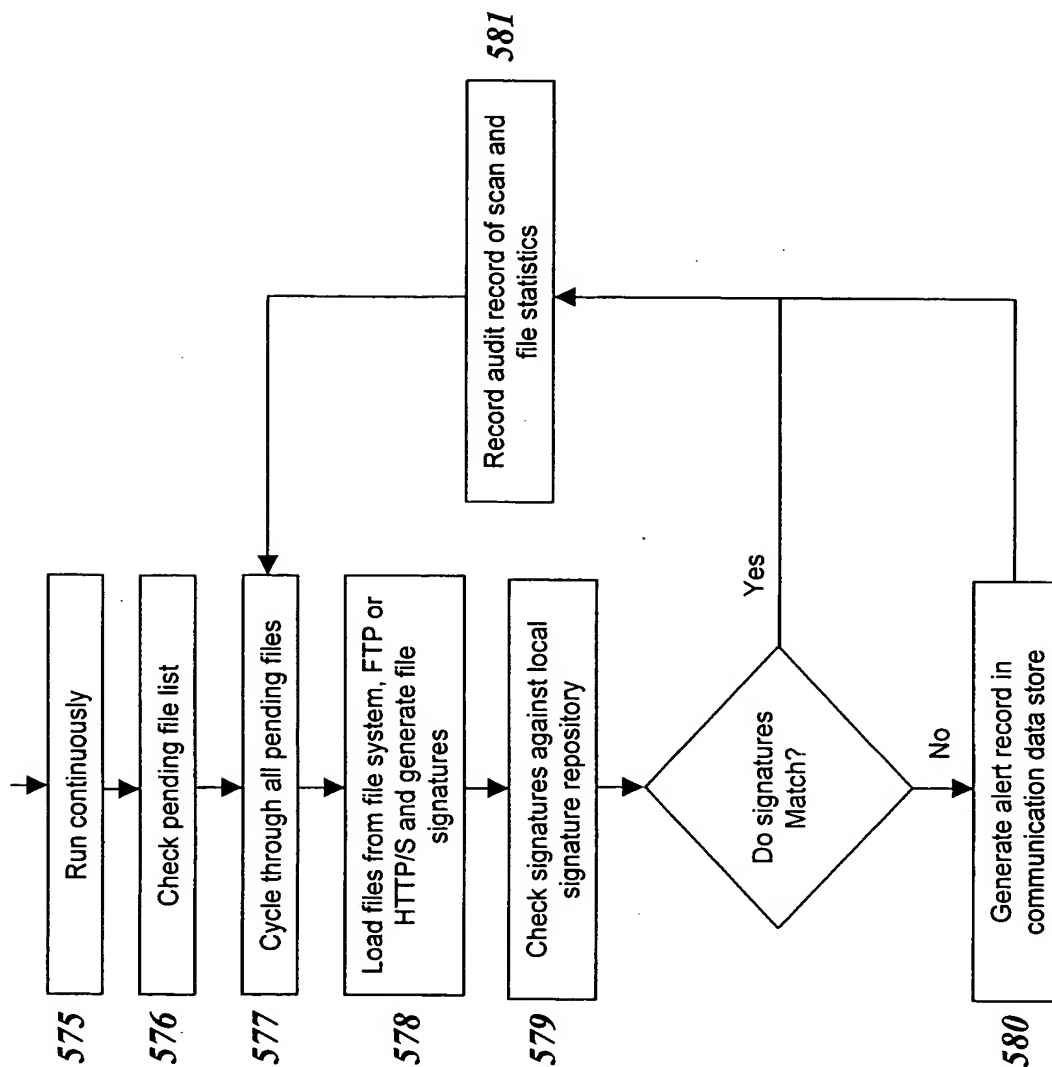


Figure 26: OnSite File Communications (on customer server)

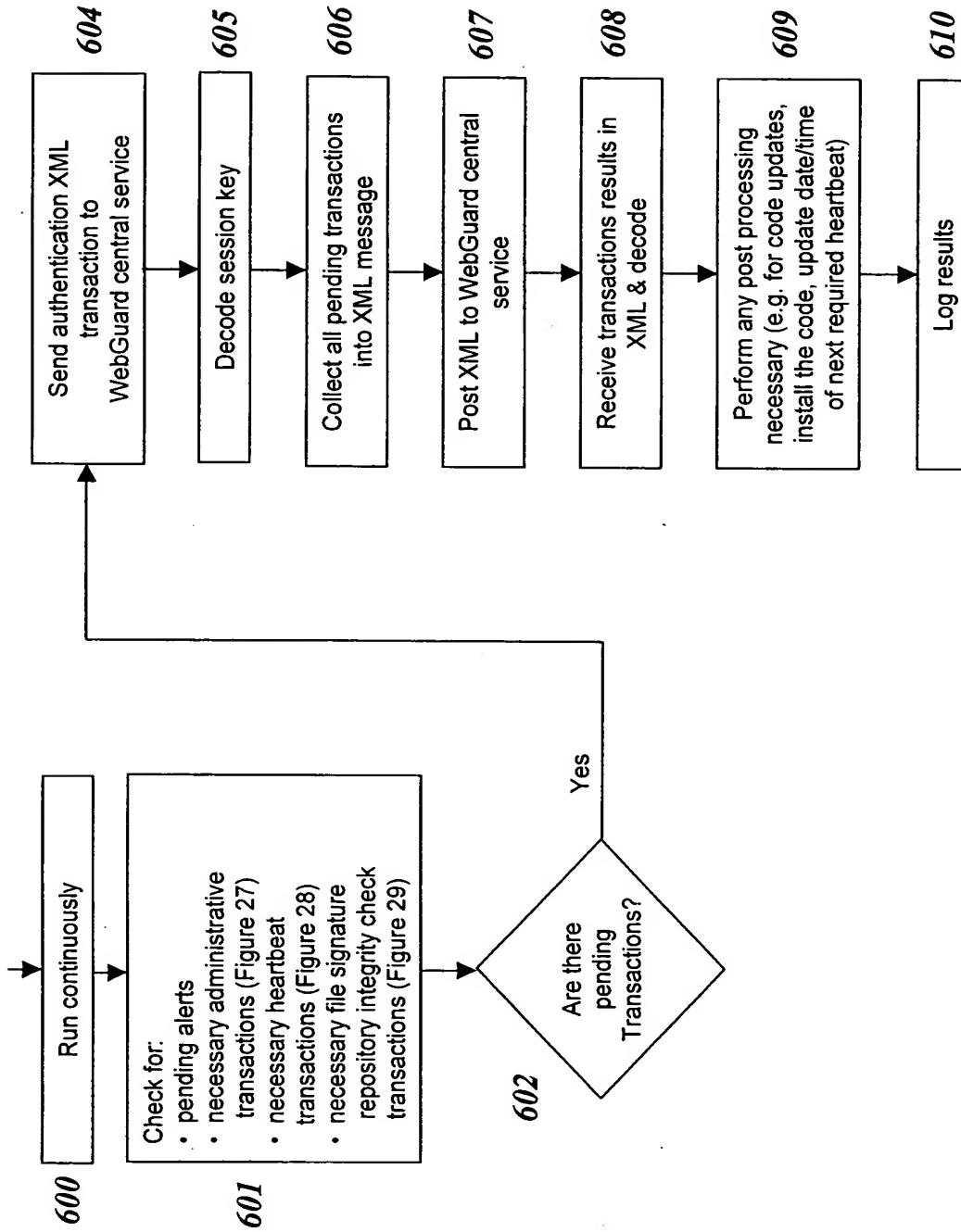


Figure 27: OnSite File Administrative Transactions

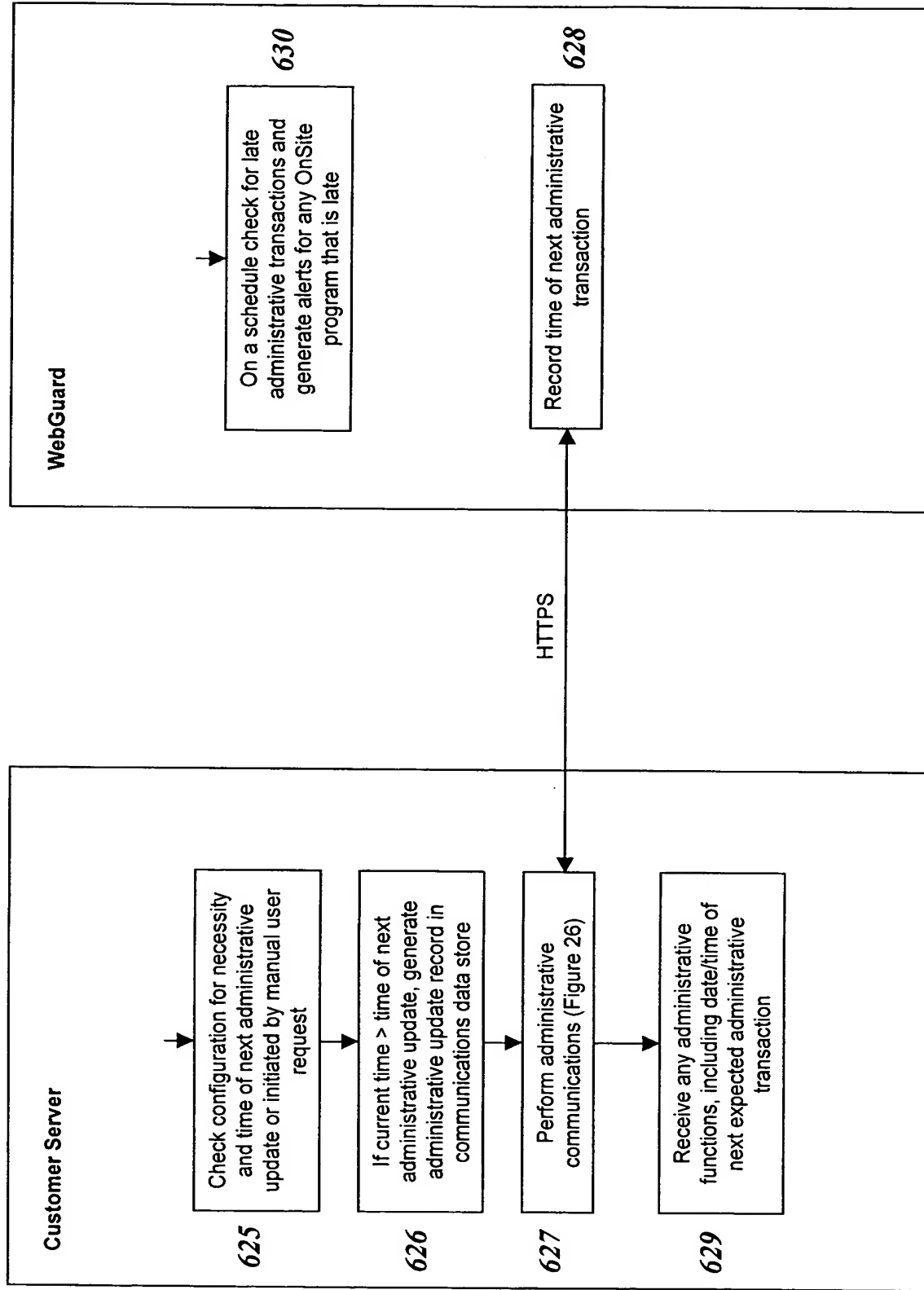
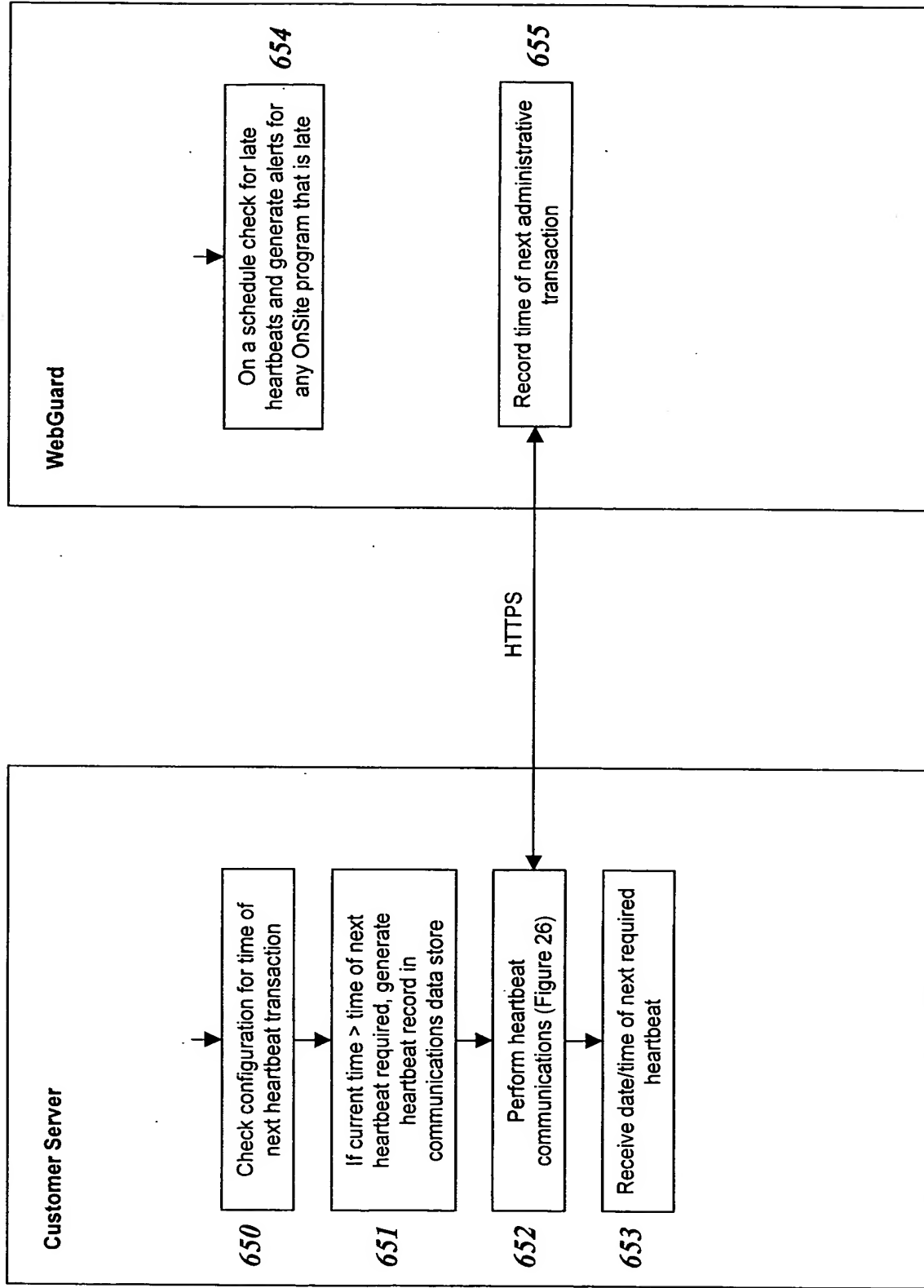
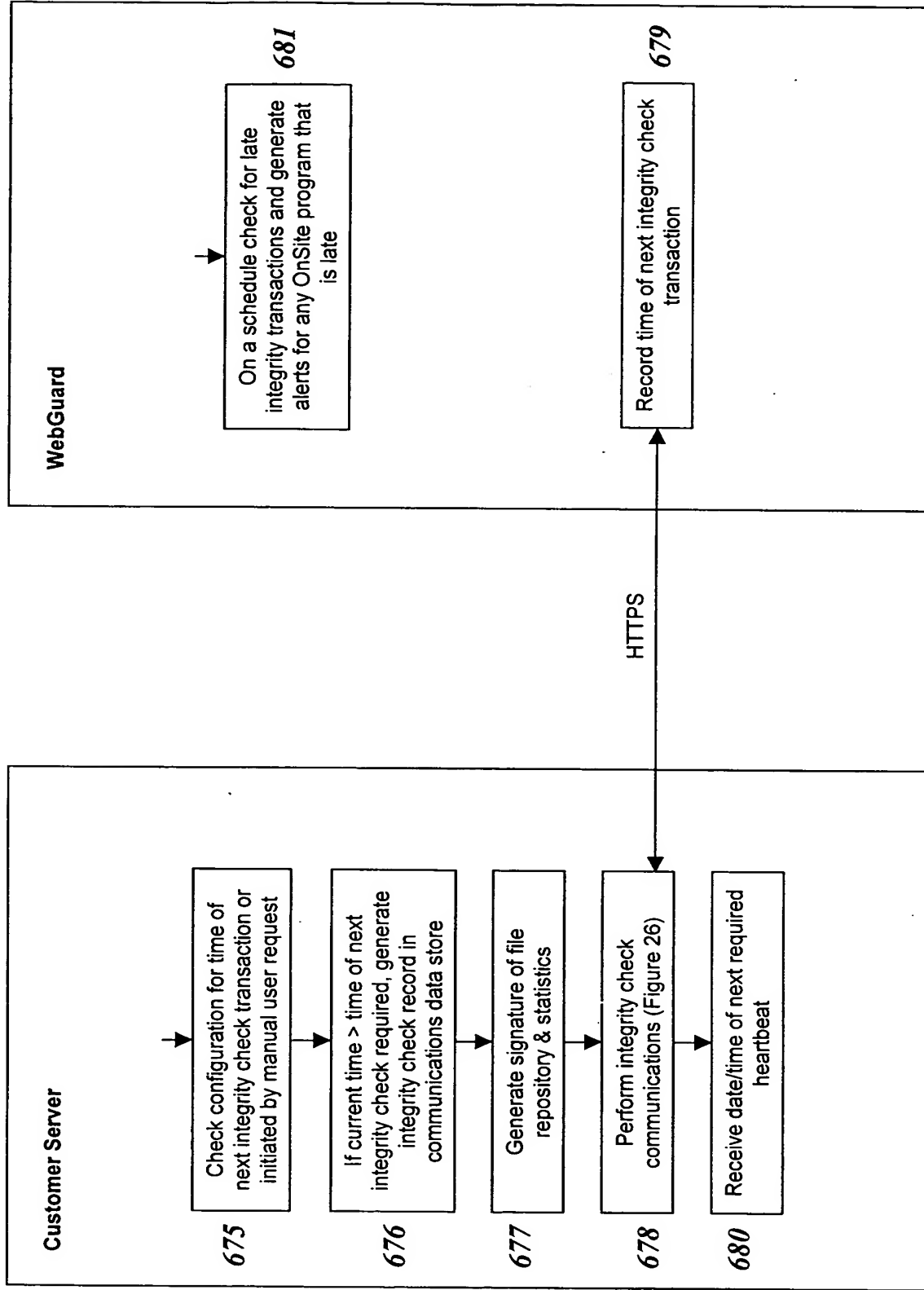


Figure 28: OnSite Heartbeat Transactions



Note: File repository integrity checks and alerts generated with WebGuard central service also count as heartbeats

Figure 29: OnSite Local File Repository Integrity Check Transactions



Note: File repository integrity checks and alerts generated with WebGuard central service also count as heartbeats

Figure 30: OnSite ServerGuard

